Comparison of Masked and Unmasked Face Recognition Performance with Hand-Crafted Methods

Ali Torbati

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Eastern Mediterranean University
February 2023
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Ali Hakan Ulusoy
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science in Computer Engineering.

Prof. Dr. Zeki Bayram
Chair, Department of Computer Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Computer Engineering.

Prof. Dr. Önsen Toygar
Supervisor

Examining Committee

1. Prof. Dr. Önsen Toygar
2. Assoc. Prof. Dr. Duygu Çelik Ertuğrul
3. Assoc. Prof. Dr. Mehtap Köse Ulukök

ABSTRACT

Face recognition, a biometric technology, identifies people by their distinctive facial attributes. An image of a person's face must be compared to a library of other people's images to authenticate their identification. Face recognition systems may be used for identification, surveillance, and safety. Deep learning algorithms provide precise face recognition even under challenging conditions. Due to COVID-19 and masks, facial identification from unconstrained images is almost impossible. To avoid COVID-19, most people use masks outside. In many cases, typical facial recognition technology is useless. The majority of contemporary advanced face recognition methods are based on deep learning, which primarily relies on a huge number of training examples. Considering simply the eye and forehead regions of the face, however, masked face recognition may be investigated using hand-crafted approaches at a lower computing cost than using deep learning systems. In this thesis, handcrafted techniques are used to extract eye region and forehead characteristics from masked faces. We intend to construct a low-cost system for recognizing masked faces and compare its performance to that of face recognition systems that do not use masks. This study compares the performance of masked and unmasked face recognition systems. Experiments are undertaken on two publicly accessible datasets for masked face recognition: Masked Labeled Faces in the Wild (MLFW) and Cross-Age Labeled Faces in the Wild (CALFW). A comparison of the performance of the systems are provided in the thesis.

Keywords: Masked face recognition, Unmasked face recognition, Hand-crafted methods

ÖZ

Biyometrik bir teknoloji olan yüz tanıma, insanları ayırt edici yüz özelliklerine göre tanımlar. Bir kişinin yüzünün görüntüsü, kimliğini doğrulamak için diğer kişilerin görüntülerinden oluşan bir veri kümesi ile karşılaştırılmalıdır. Yüz tanıma sistemleri tanımlama, gözetleme ve güvenlik için kullanılabilir. Derin öğrenme algoritmaları, zorlu koşullar altında bile hassas yüz tanıma sağlar. Kısıtlanmamış görüntülerden yüz tanıma, günümüzde COVID-19 salgını ve yüz maskelerinin yaygınlığı nedeniyle neredeyse zordur. Evlerinin dışında, neredeyse herkes COVID-19 virüsünün bulaşmasını başarılı bir şekilde sınırlamak için maske takmıştır. Bu, standart yüz tanıma teknolojilerini birçok durumda neredeyse değersiz hale getirmiştir. Çağdaş gelişmiş yüz tanıma yöntemlerinin çoğu, öncelikle çok sayıda eğitim örneğine bağlı derin öğrenmeye dayanmaktadır. Bununla birlikte, yüzün sadece göz ve alın bölgeleri düşünüldüğünde, derin öğrenme sistemlerinden daha düşük bir bilgi işlem maliyetiyle el yapımı yaklaşımlar kullanılarak araştırılabilir. Bu tezde, maskelenmiş yüzlerden göz bölgesi ve alın özelliklerini çıkarmak için el işi teknikler kullanılmıştır. Maskeli yüzleri tanımak için düşük maliyetli bir sistem kurmak ve performansını maske kullanmayan yüz tanıma sistemleriyle karşılaştırmak amaçlanmıştır. Bu çalışma böylece maskeli ve maskesiz yüz tanıma sistemlerinin performansını karşılaştırmaktadır. Maskeli yüz tanıma için halka açık iki veri kümesi üzerinde deneyler yapılmıştır: Vahşi Doğada Maskeli Etiketli Yüzler (MLFW) ve Vahşi Doğada Çapraz Yaş Etiketli Yüzler (CALFW). Tezin sonunda iki sistemin performansının bir karşılaştırması sağlanmıştır.

AnahtaKelimeler: Maskeli yüz tanıma, Maskesiz yüz tanıma, El yapımı yöntemler

I dedicate this research to, all the people in my life who touched my heart.

ACKNOWLEDGMENT

First and foremost, I would like to express my sincerest thanks to my supervisor, Prof. Dr. Önsen Toygar, for her unconditional support during the entirety of my thesis research and study, as well as for her boundless enthusiasm, patience, and immense knowledge.

I owe a deep debt of gratitude to my mother for her generous support she provided me throughout my entire life, this wouldn't be possible without her unconditional love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CALFW	Cross-Age Labeled Face in the Wild Face Image Dataset
CNNs	Convolutional Neural Networks
DNNs	Deep Neural Networks
FAR	False Acceptance Rate
FERET	The Facial Recognition Technology Face Image Dataset
FRR	False Rejection Rate
HOG	Histogram of Oriented Gradients Feature Extraction Method
LBP	Local Binary Pattern Feature Extraction Method
LDA	Linear Discriminant Analysis
ML	Machine Learning
MLFW	Masked Labeled Face in the Wild Face Image Dataset
OCR	Optical Character Recognition
PCA	Principal Component Analysis Feature Extraction Method
ResNets	Deep Residual Networks
RNNs	Recurrent Neural Networks
SDLA	Semi-Supervised Discriminant Local Analysis
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features

Chapter 1

INTRODUCTION

1.1 Why Computer Vision Is Hard

The study of artificial intelligence known as "computer vision" involves developing algorithms and models that can allow computers to interpret, analyze, and understand visual data from the world around them. It involves techniques for analyzing, processing, and understanding images and video data, as well as techniques for recognizing patterns, objects, and features in this data. Applications of computer vision include image and video processing, object recognition, and scene understanding.

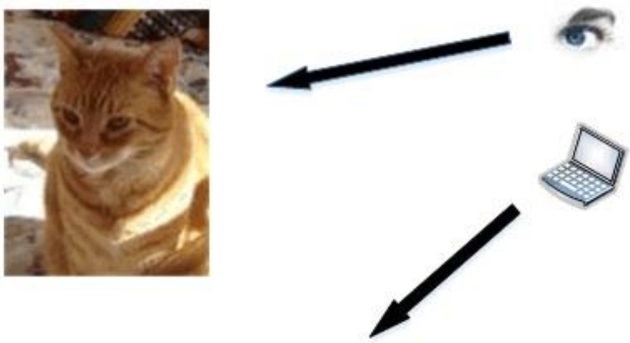
Computer vision techniques are frequently employed in the realm of image processing for analysis and editing of images. This can include tasks such as image enhancement, the purpose of which is to make an image look better, visually, and image restoration, in which the goal is to remove noise or other distortions from an image. Other common tasks in image processing include image segmentation, in which the aim is to divide the image into several segments, and object recognition, in which the goal is to identify and classify objects within an image. Computer vision techniques can also be used for 3D reconstruction, which involves generating a 3D model of an object or scene from multiple 2D images.

There are some key differences in the way that computers and humans process images. One major difference is that computers can process images much more quickly and accurately than humans, making them well-suited for tasks such as analyzing large datasets of images or detecting patterns and features in images. However, humans are generally better at understanding and interpreting the meaning and context of images, and at making decisions based on this understanding.

One of the key ways that computers process images differently from humans is by using algorithms to analyze and interpret the data in an image. Images may be analyzed by these algorithms in order to pick out particular forms, patterns, or attributes, and to use this information to classify or recognize objects within the image. In contrast, humans rely more on their visual system and their ability to understand and interpret the context and meaning of an image.

Imagine you are given an image of a cat and a dog, and asked to identify which is which. A computer might approach this task by analyzing the pixel values in the image and looking for patterns or features that are characteristic of cats or dogs. For example, the computer might look for patterns of fur, whiskers, or certain body shapes that are commonly associated with cats or dogs. Figure 1 demonstrates how a computer vision program perceives an object and how a human perceives the same object, such as a cat. On the other hand, a human might approach this task by looking at the overall context of the image and using their understanding of the characteristics and behaviors of cats and dogs to make a decision. For example, a human might look at the size and shape of the animals, their posture and expressions, and any other clues in the image that might help them understand the context and meaning of the image.

Overall, while a computer might be able to analyze an image more quickly and accurately than a human, a human might be better at understanding and interpreting the meaning and context of the image, and at using this understanding to make a decision. The strengths of computers and humans complement each other when it comes to image processing, and combining the two approaches can often lead to the best results.



51	52	54	54	67	69	72	76	81	80	77	72	59	50	44	33	26	31
58	51	49	50	51	65	72	81	82	85	81	84	80	73	64	56	43	33
60	58	53	49	50	57	73	80	84	88	86	90	86	84	77	68	57	44
66	66	58	52	52	54	71	75	83	91	92	92	87	88	83	75	68	54
80	76	68	60	59	59	69	72	83	90	95	96	91	91	86	80	77	63
89	84	77	73	69	68	68	74	82	87	95	98	95	96	87	85	83	69
94	93	85	85	79	78	78	76	80	87	95	100	98	97	95	91	88	77
96	92	89	89	85	86	88	84	87	90	93	99	99	97	97	93	92	82
98	91	88	85	85	87	91	94	96	95	93	96	99	99	98	96	95	90

Figure 1: Human vs. Computer Vision [1].

1.2 Face Recognition Process

Computers may use a technique called face recognition to identify and authenticate people in images or streaming videos. It is based on the ability of computers to analyze and understand visual data, and it involves developing algorithms and models that can extract and analyze features from images of faces.

When it comes to recognizing faces, one of the major obstacles is being able to accurately identify and distinguish one face from another, even when there are variations in lighting, pose, expression, and other factors. To achieve this, face recognition systems typically use a combination of machine learning algorithms and image processing techniques to analyze and compare the unique features of different faces.

Face recognition technology has advanced to the point where it can be trained on a huge collection of images and then utilized in real time for authentication and identification purposes. This can have a variety of applications, including security and surveillance, social media, and online authentication.

1.3 Face Collection

A face recognition system's face collection is the dataset of images used to teach the system how to detect and discriminate between individual faces. This dataset is typically compiled by gathering several images of various people, and then putting a name on every image to indicate who is shown.

The face collection is a crucial component of a face recognition system, since the performance of the system is affected directly by it. To achieve high accuracy, the face collection should be diverse and representative of the types of faces that the system will encounter in real-world scenarios. This may involve including a wide range of ages, ethnicities, and genders, as well as variations in lighting, pose, expression, and other factors.

The size of the face collection can also impact the performance of the system. In general, larger face collections tend to lead to better performance, as they provide the system with more data to learn from. However, very large face collections can also be computationally expensive to process, and may require specialized hardware or infrastructure to handle the data.

1.4 Face Detection

The term facial detection refers to the method used to search for and locate human faces inside a digital image or video frame. It is a crucial stage in the process of face recognition, as it allows the system to identify the region of the image that contains the face, and to focus its analysis on this region. There had been a variety of face detection algorithms available prior to 2001, but the Viola-Jones' work of "Rapid Object Detection using a Boosted Cascade of Simple Features" [2] - [3] represented a significant leap forward in the field. Viola-Jones developed a Haar-classifier technique, which used Haar-like characteristics instead of pixel analysis, to detect human faces.

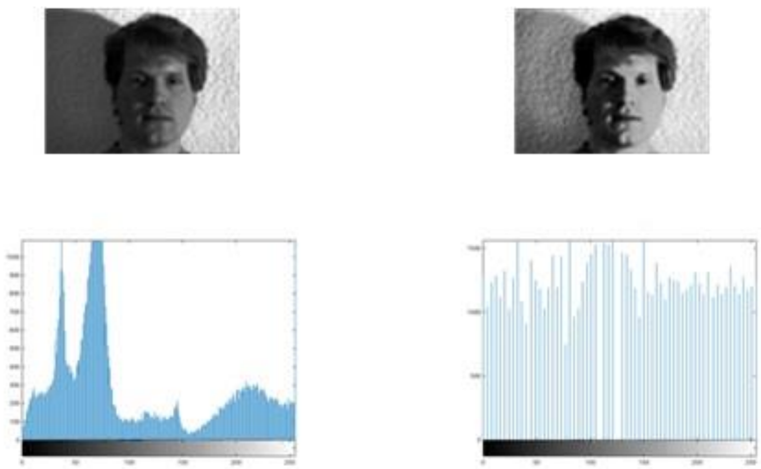
Face detection may be accomplished using a variety of methods, including machine learning algorithms, pattern recognition, and feature extraction. Identifying and localizing faces in images is a breeze with the help of the aforementioned methods, which accomplish this by studying patterns and characteristics that are unique to human faces, including the form of the eyes, nose, and mouth, as well as the structure and symmetry of the face as a whole.

Once a face has been spotted in an image, the face recognition process generally involves extracting characteristics from the face and comparing them to a database of known faces to discover a match. For best results, face detection is best utilized in tandem with other forms of image processing, namely image alignment and noise reduction, to ensure that the input image is suitable for feature extraction and comparison.

1.5 Pre-Processing

Preprocessing in face recognition refers to the process of preparing the input image or video for face detection and recognition. This may entail a wide range of activities, such as converting the image to grayscale, aligning the face images, and removing noise. Figure 2 demonstrates an example of pre-processing using histogram equalization that light levels in an image from the Yale database [4] are averaged using the histogram to provide a more consistent look. Figure 2 (a) shows the image without equalization and Figure 2 (b) demonstrates the image after equalization.

Preprocessing is beneficial to elevate the precision and accuracy of the dataset obtained, and to make it easier for the face recognition system to detect and recognize faces in the image. Some common preprocessing techniques include image alignment, noise reduction, color conversion, histogram equalization. Overall, the goal of preprocessing is to enhance the quality of the input data and to make it easier for the face recognition system to detect and recognize faces in the image.



(a) (b)

Figure 2: Pre-Processing Using Histogram Equalization

1.6 Machine Learning

Machine learning's ultimate aim is to make it so computers can teach themselves new skills by analyzing existing data and make inferences or choices without being explicitly programmed. It's predicated on the premise that computers can analyze data, spot trends, and decide what to do next.

One may classify machine learning as either supervised or unsupervised. In supervised learning, the proper results for each example in the training set are labelled and used to train a machine learning model. Lin et al. [5], presents a supervised learning approach for face recognition, in which with the help of a tagged dataset of face images, a machine learning model is trained to recognize and classify new faces. The idea is to use the learnt patterns to create predictions about novel, unseen cases. In unsupervised learning, the model is left to discover connections and patterns in the data on its own. Mian et al. [6], presents an unsupervised learning approach for video-based face recognition, in which an unlabeled dataset of face images is used to train a machine learning model to recognize the faces based on similarities in the extracted local features.

There are also other types of machine learning, such as semi-supervised learning and reinforcement learning, which combine elements of supervised and unsupervised learning. Ling et al. [7], presents a semi-supervised learning approach for face recognition with the use of discriminant local analysis (SDLA), in which in order to train a machine learning model, both labelled and unlabeled images of faces are used to recognize and classify new faces. Also, Yadav et al. [8] presents a deep reinforcement learning approach for face detection, where a machine learning model may learn to recognize people in images by observing its surroundings and responding accordingly with incentives or consequences. The suggested technique effectively detects and identifies faces in videos using deep reinforcement learning and K-Nearest Neighbors (KNN).

On the subject of face recognition, machine learning means the employment of algorithms and statistical models that can learn from data to magnify the efficiency and exactness of the system that recognizes faces.

Overall, machine learning is a potent technology for automating decision-making and solving complex problems, and it may be used in many different contexts, including speech and image recognition, processing of natural languages, and forecasting models. Furthermore, machine learning can be an effective approach for face recognition, as ultimately, this facilitates the system's ability to learn from its experiences and improve in competence.

1.7 Algorithms

There are several types of machine learning algorithms that can be used for face recognition. In order to train supervised learning algorithms, the data used must be labelled so that the proper response may be determined for each training sample. The idea is to use the learnt patterns to create predictions about novel, unseen cases. Examples of supervised learning algorithms include Linear Regression [9], Logistic Regression [10], and Support Vector Machines [11].

Unsupervised learning algorithms are not given any labeled training examples and must find patterns and relationships in the data on their own. Methods like k-Means

Clustering [12] and Principal Component Analysis are examples of unsupervised learning algorithms.

The algorithms of reinforcement learning acquire knowledge through experience with the world and the subsequent allocation of rewards and punishments. Overtime, the aim is to learn a policy that increases the total compensation. Examples of reinforcement learning algorithms include Q-learning [13] and SARSA [14].

A kind of machine learning known as "deep learning" employs several layers of artificial neural networks to discover commonalities and distinctions in the data they are fed. They are ideally suited for activities including image and voice recognition, and provide cutting-edge results across a broad range of tasks. Convolutional Neural Networks and Recurrent Neural Networks are two examples of deep learning methods.

Overall, different machine learning algorithms may be more or less suitable for face recognition depending on the characteristics of the dataset and the specific requirements of the application.

The following subsections describe some algorithms that are being used in this thesis for face recognition.

1.7.1 Principal Component Analysis

Principal Component Analysis (PCA) is a computational method for reducing the dimensionality of a dataset. How it achieves this is by projecting the data onto the directions in the data (called "principal components") that capture the highest variation.

PCA is a type of unsupervised learning algorithm, as it does not need any marked training examples. Typically, it is the first stage in the preprocessing pipeline for other machine learning algorithms, as it may minimize the data's complexity and increase the model's performance. Coots et al. [15] used PCA to produce a statistical model for locating landmarks in the training images. Rani et al. [16] proposed a face recognition program via eigenfaces for recognition, it opens the door if the face is identified and sends an email if the face is not recognized.

In the context of face recognition, PCA may be used to determine which characteristics of the facial images best represent the most significant differences in the data. A machine learning system may take these traits as input and use them to categorize faces or identify people. PCA can also be used to visualize the relationships between the faces in a dataset, by projecting them onto a lower-dimensional space.

Overall, PCA is a neat tool in the case of dimensionality reduction of a dataset and identifying the most important features or patterns in the data.

1.7.2 Local Binary Patterns

Local binary pattern (LBP) is a texture descriptor used for image analysis and object recognition. It operates on the principle of comparing the values of individual pixels in an image to a predefined threshold, and encoding the resulting binary patterns in a histogram. Kulkarni et al. [17] proposed an approach for automated face identification in real time, employing a database of previously captured images with different poses, lighting, and emotional expressions.

In the matter of face recognition, beneficial to better understand a person's face, LBP could be employed to extract characteristics from an image that accurately represent the face's local structure and texture. The data collected from these traits may be fed into a machine learning system for use in facial recognition or classification.

LBP has several advantages as a feature descriptor for face recognition. It is simple and computationally efficient, and it is resistant to lighting and position fluctuations. It is also resistant to noise and other distortions, making it well-suited for use in real-world scenarios.

Overall, LBP may be carried out to extract texture-based attributes from images, and applications such as facial recognition, object identification and image classification have made extensive use of this technique.

1.7.3 Histogram of Oriented Gradients

Object identification and detection rely on feature descriptors like the Histogram of Oriented Gradients (HOG). It is based on the idea of calculating the gradient's dispersal directions in a sample, and encoding this information in a histogram.

In the common era of facial recognition, HOG has the potential to extract features from images of faces that capture the structure and contour of the face. A machine learning program may take this information as input and use it to categorize faces or identify persons. Ahamed et al. [18] envisioned a facial detection system using HOG enabling instantaneous facial recognition and authentication.

HOG has several advantages as a feature descriptor for face recognition. It can withstand shifts in perspective and illumination, as well as noise and other aberrations. Because of its low computational cost, it is ideal for usage in real-time settings.

Overall, HOG is a useful tool for extracting shape-based features from images, and there are several uses for it, including pedestrian detection, object recognition, and image classification.

1.8 Datasets

In the topic of facial recognition, datasets are collections of images and associated labels that are used for the process of training, proving, and re-checking machine learning models. The images in a dataset typically consist of faces that have been captured under a variety of conditions, including in a variety of lighting scenarios, poses, and backgrounds. Usually, the image's label will contain the subject's identity, as well as additional information such as the gender, age, or facial expressions of the person.

Numerous studies on facial recognition have made use of these datasets and they have played a major role in advancing the state-of-the-art. They provide researchers with a standardized way to evaluate the performance of their models, and allow for fair comparisons between different methods.

It is worth mentioning that research difficulties exist and the ways to handle variations such as changes in lighting, poses, and expressions, as well as variations in age and ethnicity should be considered. This can be achieved by having a diverse dataset with a good representation of these variations. The datasets that are being used in this thesis are mentioned in the following subsections.

1.8.1 Cross-Age Labeled Face in the Wild (CALFW)

The performance on the Labeled Faces in the Wild (LFW) database, serves as a standard for unconstrained face verification systems, approaches 100% thanks to big data driven machine learning techniques. However, they contend that such precision may be overstated. Another issue is inter-generational facial dissimilarity in face identification although LFW does not give it a lot of thought, alongside diverse stances, illuminations, occlusions, and emotions. In order to include the natural intra-class variance introduced by the ageing process, they build a Cross-Age LFW (CALFW) [19] by searching for and selecting 3 thousand positive face pairings with age gaps. To further mitigate the impact of attribute differences across positive/negative pairs, they additionally pick negative pairs that share the same racial and gender composition. CALFW database has 6,000 face pairings (Distribution of 50%-50% of positive and negative face pairings).

1.8.2 Masked Labeled Face in the Wild (MLFW)

It is possible that current face recognition algorithms suffer significant performance drops while trying to identify masked faces, as an increasing number of people choose to protect themselves from the current COVID-19 epidemic by putting face coverings. Using the Cross-Age LFW (CALFW) database as a foundation, they developed a straightforward yet successful technique to make masked versions of given faces automatically, and they created a new database called Masked LFW (MLFW) [20] to analyze the influence of masks on facial recognition models. The masked face that is created using their process looks quite similar to the actual face, thanks to the high quality of the mask tool. To generate a wide range of generative effects, they also gather a library of mask templates encompassing the vast majority of styles encountered in everyday life that can be seen in Figure 3. The MLFW database is generated with the use of 31 mask templates and the aforementioned parameters. For each face image in the MLFW database, they provide the 250 x 250-pixel masked image and the alignment landmarks.

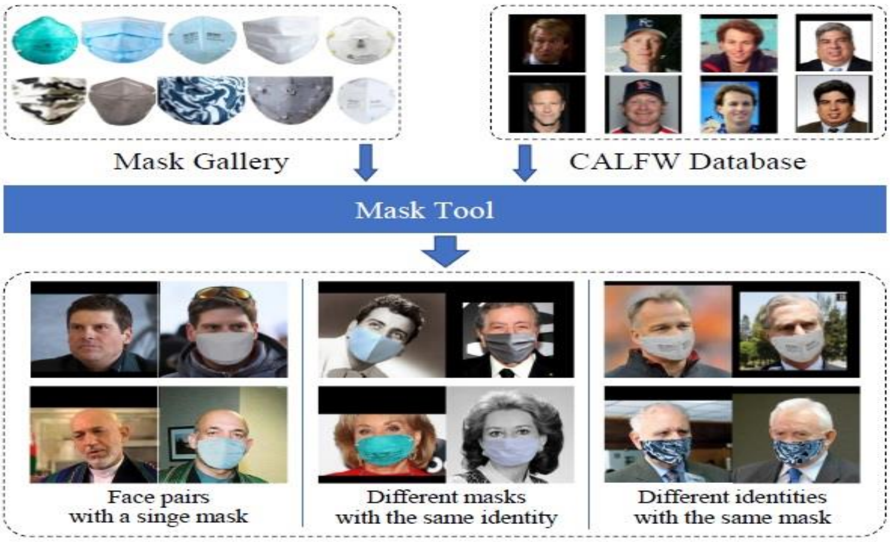


Figure 3: Masked LFW Dataset Development [20]

1.8.3 The Facial Recognition Technology (FERET)

To assess the performance of diverse facial recognition systems, researchers have created the Face Recognition Technology (FERET) database [21]. Dr. Jonathan Phillips of the Army Research Laboratory in Adelphi, Maryland, and Dr. Harry Wechsler of George Mason University founded it in 1993. The FERET dataset is a standardized set of face images employed for development of algorithms and the publication of related findings. Having access to a centralized database also made it possible to evaluate the relative merits of various methods. Between December 1993 and August 1996, 14,126 images representing 1199 persons were gathered for the database, along with 365 sets of duplicate images shot on separate days.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

Face recognition is a rapidly evolving section that has attracted significant attention from researchers and practitioners in recent years. It is a technique for identifying and verifying individuals based on their facial features, and it may be used in a variety of contexts, including security, surveillance, biometrics, and human-computer interaction. The goal of face recognition is to accurately match an input face image to a pre-existing template, or to some number of templates.

The process of face recognition is composed of several stages, including face detection, alignment, feature extraction, and matching. Face detection is used to locate the face in the image, alignment is used to correct for variations in pose and lighting, extracting useful features from a face is called feature extraction, and matching is used to compare the input face to the templates. Accuracy and failure rate are common ways of gauging a face recognition system's effectiveness.

Face recognition systems' performance has dramatically improved indeed for latest innovations in deep learning. Deep learning algorithms, like Convolutional Neural Networks, have demonstrated excellent performance on a wide range of face recognition benchmarks. However, there are still challenges to overcome in the field, such as dealing with variations in lighting, pose, and expression, and handling large-scale datasets.

This literature review's intention is to provide an abstract of recent and significant studies with regard to facial recognition, focusing on the different methods used, the evaluation metrics and benchmarks, and the challenges and trends in the field. The review will cover both traditional and deep learning-based approaches to face recognition and it will discuss their strengths and weaknesses. The study will provide light on current knowledge gaps and suggest future research avenues.

2.2 State-of-the-Art

Facial recognition technology has advanced greatly in recent years thanks to the use of machine learning methods. One of the most popular and effective approaches is the use of deep neural networks (DNNs), including convolutional neural networks and deep residual networks (ResNets). It has been shown that these models can perform at the cutting edge on many facial recognition criteria, such as the Labeled Faces in the Wild (LFW) dataset and the MegaFace dataset.

One of the recent trends in face recognition research is the development of methods for recognizing faces in the presence of masks. As a result of the COVID-19 epidemic, masks have become commonplace and face recognition systems need to be able to handle this new variation of faces.

The issue of masked facial recognition may be tackled from two different angles. One is based on generative models that generate synthetic faces of individuals wearing masks, and the other is based on modifying the existing recognition systems to handle masked faces.

Generative methods such as Generative Adversarial Networks and Variational Autoencoders have been used to generate synthetic masked faces, which are then used to fine-tune the existing recognition systems. These methods have shown promising results in terms of the recognition rate, but there is still room for improvement. Another approach is to modify the existing recognition systems, such as by adding a separate branch to the network to handle the masks or by using attention mechanisms to focus on the unmasked regions of the face. In terms of recognition rate, these strategies have shown encouraging outcomes, but there are still challenges in dealing with occlusions and variations in the mask.

Overall, the current technological status in face recognition has been continuously advancing, and with the assistance of machine learning methods and deep learning in particular, it's achieving excellent performance. However, the recent challenges of masked-face recognition have put a spotlight on this area, and researchers have been working on developing methods to handle this new variation of faces with promising results, but still, it is an ongoing area of research.

2.3 Methods

The existing research presents a number of different methods for face recognition. There are three primary groups into which these methods fall as feature-based, template-based and deep learning-based methods that are described in the following subsections.

2.3.1 Feature-Based Methods

They can extract features from the face image that are used to represent the face and then use these features to compare the input face to the templates. Examples of feature-based methods include the Eigenface method [22], the Fisherface method [23], and the

Local Binary Pattern (LBP) method [24]. These methods have been widely used in the past but their performance is limited by the variations in pose, lighting and expression.

2.3.2 Template-Based Methods

Template-based solutions, store the entire face image as a template and directly compare the input face image to the stored templates. These methods include methods such as the Nearest Neighbor method [25], the Mahalanobis distance method [26], and the kernel-based method [27]. These methods have shown to be resistant to changes in posture and lighting but are still limited by the variations in expression.

2.3.3 Deep Learning-Based Methods

These techniques figure out how to map the facial images utilizing DNNs to a feature space where the similarity between faces can be measured. On a variety of face recognition benchmarks, these strategies have shown to perform at the state-of-the-art level. Examples of these methods include deep convolutional neural networks (DCNNs) [28], deep residual networks (ResNets) [29], and MobileNet [30].

2.4 Evaluation

Evaluation is a critical aspect of face recognition research as it allows researchers to measure the performance of their methods and compare them to existing approaches. There are several metrics and benchmarks commonly used in the literature in order to assess how well facial recognition systems works.

One commonly used metric is the recognition rate, or accuracy, which is the amount of test images that are properly recognized. Another ordinarily used metric is the error rate, that is misdiagnosis rate in a test set of images. These metrics are typically reported as a function of the False Acceptance Rate (FAR) and the False Rejection

Rate (FRR), which are the chances of mistaking a fake user for the real deal and vice versa, respectively.

A popular benchmark for facial recognition is the Labeled Faces in the Wild (LFW) dataset [31], that includes over 13 thousand images of faces gathered from throughout the web and tagged with the name of the individual depicted. Many facial recognition method comparisons rely on this dataset, as it is large and challenging, with a lot of room for variation in terms of lighting, pose and expression.

Another benchmark that has been proposed for testing facial recognition in real-world scenarios is the MegaFace dataset [32], which is a million-scaled dataset with more than 690,000 images of more than 530,000 individuals. The dataset provides a realistic and challenging test bed for face recognition methods and allows researchers to evaluate performance at scale.

Recent research also focuses on the evaluation of face recognition performance on masked faces, using datasets such as the Masked Face Recognition dataset (MFSD) [33]. These datasets make it possible to test how well detection techniques do with this new type of masked facial image.

2.5 Conclusion

It is clear that, in recent years, researchers and businesspeople have devoted considerable effort to the field of facial recognition. Accurately matching an input face image to a pre-existing template, or group of templates, is the objective of face recognition. The process of recognizing a person's face involves a number of steps, the most important of which are face detection, alignment, feature extraction, and finally, matching.

Various methodologies have been presented and assessed in the literature throughout the years. There are essentially three basic groups into which these techniques fall: feature-based methods, template-based methods, and deep learning-based methods. Recent deep learning advancements have greatly increased accuracy of face recognition systems. Methods based on deep neural networks like convolutional neural networks and deep residual networks [29] have demonstrated excellent performance on different facial recognition standards such as the Labeled Faces in the Wild (LFW) dataset [31] and the MegaFace dataset [32].

On the other hand, a recent trend in face recognition research is the development of methods for recognizing faces in the presence of masks. This is a challenging problem due to the occlusions caused by the masks, and various methods based on generative models and models based on deep learning have been suggested to address this problem. Datasets such as the Masked Face Recognition dataset (MFDD) [33] have been introduced to evaluate the performance of these methods.

To wrap up, beneficial to accomplish the complex task of face recognition, some methods and strategies were offered in the scientific literature. The field is still evolving, with new challenges and trends arising, such as the recognition of masked faces. Future work should continue to focus on addressing these challenges, improving the performance of face recognition systems, and developing new and more effective methods.

Chapter 3

FEATURE EXTRACTION AND RECOGNITION

3.1 What is Feature

In computer vision, a feature is a quantifiable piece of data in an image that is particular to an item. It may be a specific color or shape, such as a line, edge or image section. A feature, therefore, is any piece of information that is required to finish the computing process associated with a certain application. Specific structures in an image, such as points, edges, or objects, may constitute features. The implementation of a general neighborhood operation on an image or the detection of features can also result in the creation of features in that image. A distinct characteristic distinguishes one item from another.

In machine learning projects, we must convert the raw data (images) into such a features vector to demonstrate to the learning algorithm how to fully comprehend the object's attributes. In the majority of machine vision applications, algorithms utilize features as the essential basis for recognizing differences between objects. The application's quality will rely on the feature descriptor employed. Due to its significance, feature extraction requires a great deal of effort [34].

Early steps usually involve preprocessing the images. Better features may be derived from an image if the image itself is improved by preprocessing. Image preprocessing often entails tweaks to the adjusting contrast, saturation, hue, histogram equalization, and thresholding.

In order to classify and recognize the images, the next step is to extract features. It is not advisable to put all of your faith into a single feature extractor, as some may perform well in some scenarios but poorly in others. Therefore, selecting an effective extractor during preprocessing will boost the performance of the final application. High-quality features tend to lead to high-performance applications.

The data volume is also a concern. Selecting a quick and accurate feature extractor is essential if your dataset is reasonably large. Because processing a huge database necessitates a lot of memory and also increases computation time, improved results may be expected if the amount of input data is minimized. Two categories can be used to classify features [35], namely general features and domain-specific features that are described in the following subsections.

3.1.1 General Features

Texture, color, and shape all contributed to the formation of the image. It is possible to segment it further into the following categories:

- Pixel-level attributes are characteristics of a single pixel, such as its color and position among other factors.
- Local features are the features that are obtained when the image is being segmented or while edge detection is taking place.
- Features that are retrieved from the whole image or a tiny portion of it are called global features.

3.1.2 Domain-specific Features

Localized identifiers, such as fingerprints and eyes, are being used in a variety of contexts. Low-level features and high-level features are the main categories into which all other features fall. Simple features are taken straight from the images. On the other hand, high-level characteristics emerge from lower-level ones. Combining local and global characteristics requires a novel approach. Attributes using a tree structure can be combined with two layers. Thus, as shown in Figure 4, it is evident that the global attributes are saved at the top, or "root," level of the tree, while the local features are kept in the "child" nodes.

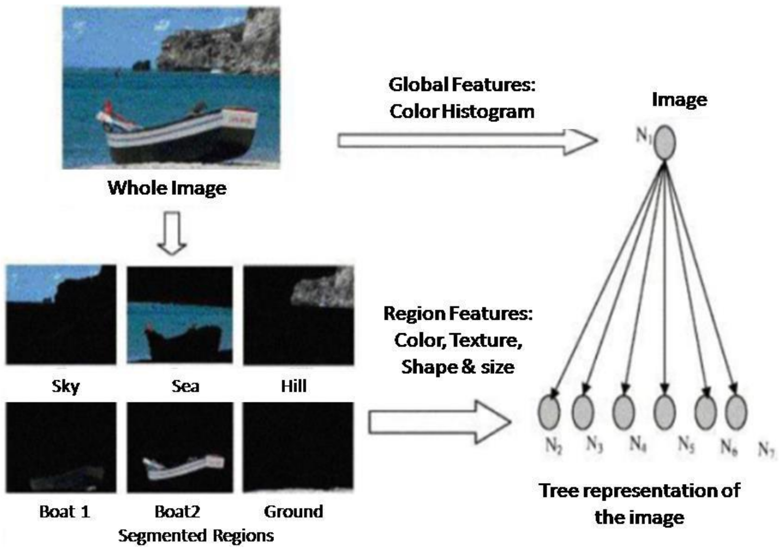


Figure 4: Tree Representation of Image Features [35].

3.2 Feature Representation

In face recognition, feature representation involves extracting a set of salient and distinctive identifying characteristics extracted from an image of an individual face. These features are chosen for being robust to variations of lighting, pose, and facial expressions, and to capture the unique characteristics of a face.

One prevalent method is to extract characteristics based on the geometric structure of the face, such as the distances and ratios between expressions of the eyes, nose, and mouth, and jawline. These features are often referred to as geometric features, and they can be used to characterize the overall shape and structure of the face. Another approach is to extract features based on the texture and pattern of the face, such as the skin tone, the wrinkles, and the distribution of facial hair. These features are often referred to as texture features, and they can be used to capture the fine details of the face.

Deep learning, a sophisticated technology that can learn to extract features from the data automatically, is another common method for feature extraction. Convolutional Neural Networks happened to be utilized extensively to extract characteristics from facial scans by training the network on a huge collection of face images, eventually, the network is able to identify patterns that are unaffected by the subject's lighting, location, or emotional state, and that are well suited for face recognition. These features are often referred to as deep features, and they can be highly effective for recognizing faces.

A fixed-length feature vector, which would be a numerical value of the image, is often used to represent features once they have been retrieved. The feature vectors may then be used to contrast several faces against each other using a distance measure [36], like Euclidean Distance [37], Cosine Similarity [38], or Mahalanobis Distance [39]. It is worth mentioning that feature representation is critical step in the pipeline of face recognition, and the choice of features has a big impact in terms of the system's overall efficiency.

3.3 Feature Extraction Methods

In general, feature extraction is the reduction of the complexity of the input images while preserving the most important information for a specific task. Feature extraction methods are utilized in a variety of disciplines, including pattern recognition, computer vision and image processing. There are many feature extraction techniques such as:

- Hand-crafted feature extraction methods that use one's own experience and knowledge to determine which traits are important for a certain job. Examples include Principal Component Analysis, Histograms of Oriented Gradients, Speeded Up Robust Features, Linear Discriminant Analysis, Local Binary Patterns, Scale-Invariant Feature Transform and Gabor Filter.
- Deep Learning-based feature extraction approaches that directly discover features from the data using deep neural networks. Convolutional Neural Networks and Recurrent Neural Networks are used often in this circumstance. Deep features relate to the features that deep neural networks learn, and they can be highly effective for various labors namely object detection, image classification and face recognition
- Hybrid feature extraction methods that are about combining traditional feature extraction with that performed by deep neural networks.

It is vital to note that the selection of the feature extraction technique relies on the particular job and the kind of data, and it is crucial to assess how well each approach performs on the particular dataset. In this study, the operations make use of three different feature extractors, each of which will be discussed in further depth in the forthcoming sections.

3.3.1 Principal Component Analysis

Principal Component Analysis [16] is a widely used method for feature extraction over several disciplines, including but not limited to pattern recognition, computer vision, and image processing. The original data are transformed using a linear method into a reduced-dimensional subspace, in which the directions with the highest variance in the data are retained and the others are discarded.

Here are the detailed steps for PCA:

1. Compute the mean of the data:

   First, the data should be examined and the average should be determined, which is used to center the data around the origin. This is typically done by computing the average of each feature across all data points as follows:

   $$Z = \frac{X-\bar{X}}{\sigma} \tag{1}$$

   and

   $$Mean = \bar{X} = \frac{\sum_{i=1}^{n} x_i}{n} \tag{2}$$

   $$Standard\ Deviation = \sigma = \sqrt{\frac{\Sigma(X-X_i)^2}{n}} \tag{3}$$

   where, X correspond to data, $\bar{X}$ is mean of data, n is number of data points and σ is standard deviation.

2. Compute the covariance matrix:

   Data covariance matrices must then be calculated, which is used to measure the correlation between different features. A square matrix with dimensions proportional to the number of characteristics makes up the covariance matrix, and its entries are defined as the covariances between different features as follows:

$$Covariance\ Matrix = \begin{bmatrix} COV(X,X) & COV(X,Y) \\ COV(Y,X) & COV(Y,Y) \end{bmatrix} \tag{4}$$

and

$$Covariance = \frac{\Sigma(\,X-\bar{X})(Y-\bar{Y})}{Number\ of\ data\ points} \tag{5}$$

where X is x axis, Y is y axis, $\bar{X}$ is mean of the data on x axis and $\bar{Y}$ is mean of the data on y axis.

3. Compute the eigenvectors and eigenvalues of the covariance matrix:

   Once the covariance matrix is computed, its eigenvalues and eigenvectors must be found. The eigenvectors are the directions in the data that have the highest variance, and the eigenvalues are the corresponding variances. The eigenvalue problem for the covariance matrix can be used to calculate these eigenvectors and eigenvalues, which can be done using standard linear algebra techniques as shown below.

$$|Covariance\ Matrix - Eigenvalues * Identity\ Matrix| = 0 \tag{6}$$

$$Covariance\ Matrix \times \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} Eigenvalue \times a \\ Eigenvalue \times b \end{bmatrix} \tag{7}$$

and

$$Identity\ Matrix = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{8}$$

$$Determinant \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = (a \times d) - (b \times c) \tag{9}$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = Eigenvector \tag{10}$$

4. The eigenvectors with the highest eigenvalues selection:

   Once the eigenvectors and eigenvalues are computed, the next step is to select the eigenvectors with the highest eigenvalues, forming a feature vector. These eigenvectors are the principal components of the data and they capture the most important information for the data. The intended dimensionality reduction, or explained variance of the data, often dictates the number of eigenvectors used.

5. Project the data into the new feature space:

   The last action is to project the data into the new feature space, defined by the eigenvectors selected in step 4. This is accomplished by calculating the data's and eigenvectors' dot product as follows:

$$ProjectedData = EigenVector * StandardizedOriginalDataset \tag{11}$$

PCA can be useful for various tasks such as data visualization, dimensionality reduction and feature extraction. Additionally, PCA can also be used to denoise the data or to find variations and structure in the data. However, it's worth noting that PCA is a linear method and it only captures linear relationships between the features, non-linear relationships may not be captured by this method, other non-linear methods such as Kernel PCA or Autoencoder should be considered if data has non-linear structure. Figure 5 demonstrates the PCA algorithm with detailed steps.

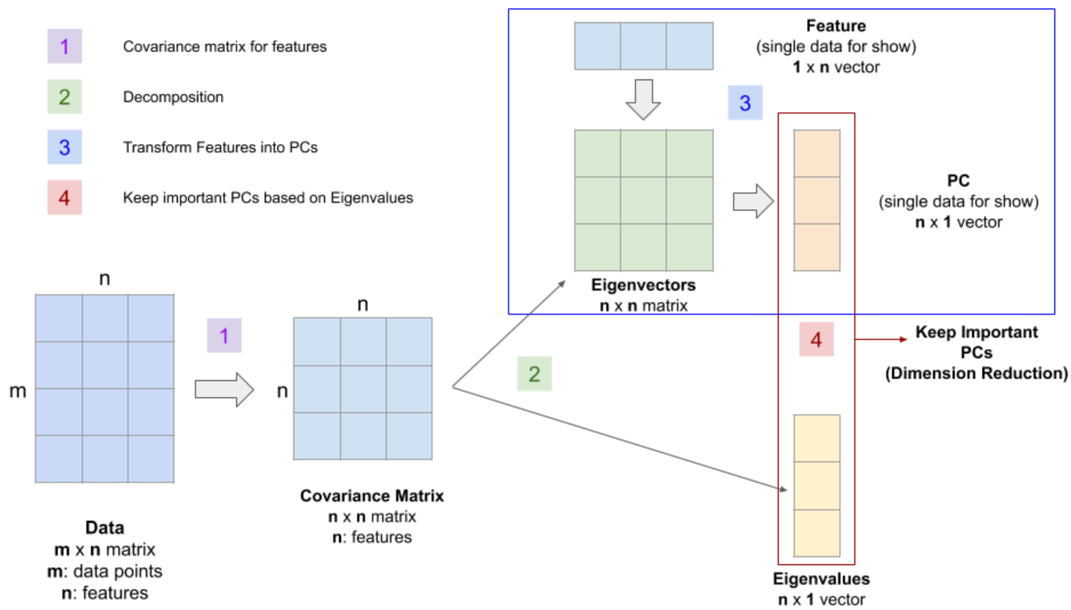


Figure 5: Principal Component Analysis Flowchart [16]

3.3.2 Local Binary Patterns

In the areas of pattern recognition, computer vision and image processing, Local Binary Patterns (LBP) [40], is implemented as a feature extraction approach. It is particularly useful for texture analysis and image classification tasks.

LBP's underlying premise is generating a binary pattern for each pixel in an image based on the surrounding pixels' intensity levels. The LBP operator contrasts a central pixel's intensity value with the intensity values of the pixels around it and the output is a binary code that represents the local texture around that pixel.

Here are the detailed steps for LBP that can be seen in Figure 6:

1. Define the neighborhood:

   The first step is to define the neighborhood of the center pixel. This is typically done by selecting a square or circular region of pixels around the center pixel. The number of pixels in the area determines how big the neighborhood is and the neighborhood is defined as follows:

   $$N = (C_x - \sin\frac{(2\pi(pixel\ index)}{8}), (C_y - \cos\frac{(2\pi(pixel\ index)}{8}) \qquad (12)$$

   where N is 3×3 neighborhood of central pixel and C is central pixel as threshold for neighbors.

2. Compute the LBP code for each pixel:

   Once the neighborhood is defined, the LBP code must be calculated for each pixel in the image as the following step. To do this, the intensity values of the core pixel are compared to those of the surrounding pixels in the immediate area. The appropriate bit in the LBP code is set to 1 if the intensity values of the neighboring pixels are greater or equal to the center pixel, otherwise, it is set to 0 as shown below:

$$LBP(gp_x, gp_y) \sum_{p=0}^{7} S(gp - gc) \times 2^p \quad (13)$$

and

$$S(x) = \begin{cases} 1 \; if \; x \geq 0 \\ 0 \; if \; x \leq 0 \end{cases} \quad (14)$$

where gc is intensity value of the central pixel and gp is intensity value of neighboring pixel with index p.

3. Perform a uniform pattern enhancement:

   Most LBP variations provide a large number of patterns, among those patterns the uniform patterns have proven to be the most discriminative. Therefore, it is often preferred to perform a uniform pattern enhancement. This is done by performing a bit-wise rotation on each of the LBP codes, and only keeping the minimum value of each code after the rotation.

4. Compute the histogram of LBP codes:

   Finally, the LBP codes are used to generate a histogram that depicts the LBP code distribution in the image. The texture of the image may be represented by this histogram, which may serve as a feature vector as follows:

$$LBP \; feature \; vector = concatenation \; of \; all \; pixels \; binary \; code \quad (15)$$

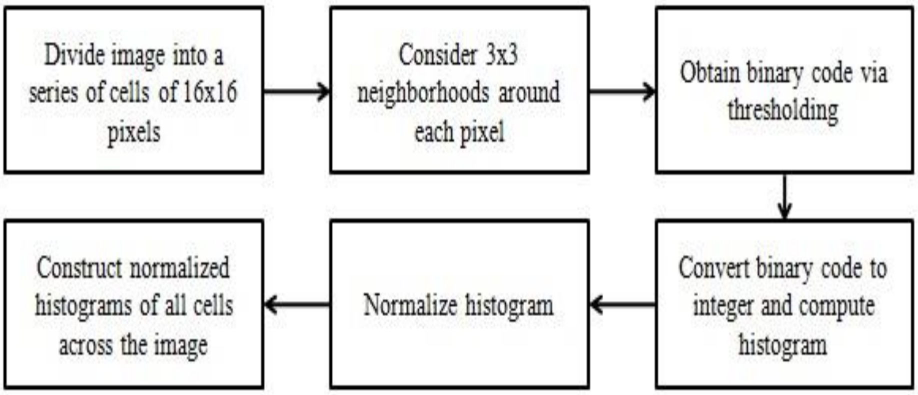


Figure 6: Local Binary Patterns Flowchart [40]

LBP has many variants, like multi-scale LBP, rotation-invariant LBP and extended LBP, and it is worth noting that LBP is used for many purposes, including but not limited to face recognition, object recognition, texture classification and image segmentation.

3.3.3 Histograms of Oriented Gradients (HOG)

The feature extraction technique known as Histograms of Oriented Gradients (HOG) [41] is widely utilized in computer vision and pattern recognition. especially for object detection tasks. The primary premise of HOG is to compute the distribution of gradient orientations in an image, which captures the shape and structure of an object.

Here are the detailed steps for HOG with an example shown in Figure 7:

1. Compute gradient magnitude and orientation:

   It all starts with figuring out how big and which way the image's gradients are. This is often performed by applying the Sobel operator to the image and determining the gradient in the x and y axes. Gradient magnitude is found by taking the square root of the total of the squares of the gradients in both the x and y axes. The gradient orientation is computed as the arctangent of the y gradient divided by the x gradient as follows:

   $$G_x(r,c) = I(r,c+1) - I(r,c-1) \tag{16}$$

   $$G_y(r,c) = I(r-1,c) - I(r+1,c) \tag{17}$$

   $$Magnitude(\mu) = \sqrt{G_x^2 + G_y^2} \tag{18}$$

   $$Orientation(\theta) = \left|\tan^{-1}(G_y/G_x)\right| \tag{19}$$

   where r and c are rows and columns respectively.

2. Divide the image into non-overlapping cells:

Once the gradient magnitude and orientation are computed, the image is then divided into a grid of separate, non-overlapping cells. Typically, the size of the item of study will dictate the size of the cells used to study it.

$$Number\ of\ bins = 9(ranging\ from\ 0° \ to\ 180°) \tag{20}$$

$$Step\ size(\Delta\theta) = \frac{180°}{number\ of\ bins} = 20° \tag{21}$$

where bin boundaries are [Δθ·j, Δθ·(j+1)] and bin center value or Cj is Δθ(j+0.5).

3. Compute a histogram of gradient orientations for each cell:

   Each cell's gradient orientations are shown in a histogram. This histogram captures the distribution of gradient orientations in the cell, which is a representation of the texture of the cell as shown below:

$$j = \left\lfloor (\frac{\theta}{\Delta\theta} - \frac{1}{2}) \right\rfloor \tag{22}$$

$$V_j = \mu \cdot \left[\frac{\theta}{\Delta\theta} - \frac{1}{2}\right] \tag{23}$$

$$V_{j+1} = \mu \cdot \left[\frac{\theta - C_j}{\Delta\theta}\right] \tag{24}$$

   where j is jth bin, Vj is the defined value for jth bin and V(j+1) is the defined value for (j+1) th bin.

4. Block normalization:

   A histogram feature descriptor is created by concatenating the histograms of all the cells, however this feature descriptor is sensitive to illumination changes, therefore it is important to normalize the descriptor over larger regions of the image. The normalization is usually done by concatenating the histograms of multiple adjacent cells and normalizing the result, often referred to as blocks as follows:

$$f_{bi} = Histogram\ of\ gradients\ for\ each\ bin \tag{25}$$

$$f_{bi} \leftarrow \frac{f_{bi}}{\sqrt{\|f_{bi}\|+\varepsilon}} \tag{26}$$

where ε is a small value in order to avoid zero division error.

5. Extract features vector:

   The histograms of all the cells or blocks are concatenated in the last stage to extract a feature vector from the image. This feature vector can be used to represent the shape and structure of the object of interest.

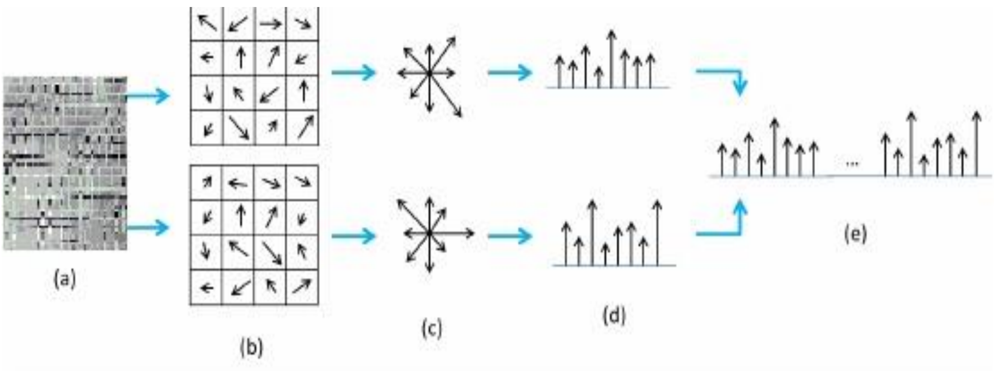


Figure 7: Histogram of Oriented Gradients [41]

It is important to point out that HOG features are both rotation and translation invariant, and it has been widely used for object recognition and face detection, it is also widely used as a feature extractor for images in the bag-of-words models [42] and also, it is a widely used feature descriptor in the pipeline of object detection using sliding window techniques [43] or CNN-based object detectors.

3.4 Recognition

Machine vision means that machines have the potential to interpret and understand visual information from the world, such as images and videos. Recognition is a vital challenge in machine vision, and it involves the ability of a machine to identify and classify objects, scenes or events in visual data.

There are different types of recognition tasks in machine vision, including:

1. Object recognition: involves identifying and classifying objects in an image or video, such as cars, faces, or animals. Various techniques may be used to conduct object recognition namely template matching, feature-based matching, or deep learning.
2. Scene recognition: involves identifying and classifying the scene in an image or video, such as an indoor scene or an outdoor scene. Scene identification may be done in a variety of ways, including feature-based matching or deep learning.
3. Face recognition: includes recognizing and validating the identity of individuals in an image or video based on their facial features. Face recognition may be achieved using numerous techniques, like as Linear Discriminant Analysis, Principal Component Analysis or deep learning.
4. Gesture recognition: involves identifying and interpreting human gestures, such as hand motions, facial expressions or body language in an image or video. Gesture recognition may be achieved using a variety of techniques, like template matching, feature-based matching or deep learning.
5. Text recognition: involves recognizing and extracting text from images or video frames, commonly used in applications like scanning invoices, forms, or license plates. Text recognition can be performed using Optical Character Recognition (OCR) techniques, which can be based on rule-based approaches or deep learning.

In general, recognition in machine vision can be seen as a process of finding patterns and structure in visual data, and it has a significant effect in a variety of applications including autonomous cars, surveillance systems and human-computer interaction.

Chapter 4

PROPOSED METHOD

Feature extraction, normalization, standardization, matching and classification are the main components of the proposed method used in this thesis. The proposed method is implemented using the training and test images, which include both masked and unmasked face images. The procedure is graphically represented in Figure 8. The adopted stages and their respective mechanisms are outlined in the following subsections.

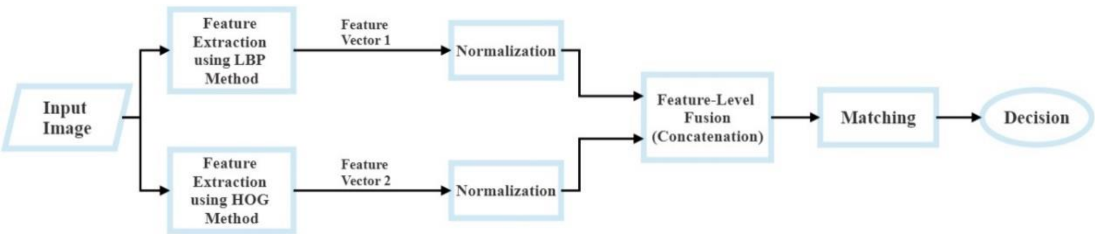


Figure 8: Block Diagram of the Proposed Method

4.1 Feature Extraction Stage of the Proposed Method

The images of people's faces are processed by the HOG and LBP techniques, in which each extract a different collection of features. Face images are mined for histogram orientation gradients in the X and Y axes using HOG, while binary patterns are mined using LBP by partitioning the images into equal-sized cells. To minimize dimensions and facilitate further processing while preserving the distinctiveness of the characteristics retrieved from the pixel values, these values are transformed to binary.

In this analysis, we employ all available face image feature extraction techniques and compare them using our training and testing data.

4.2 Normalization

Many machine learning methods use feature comparison to look for patterns in the data. But things get tricky when the sizes of the characteristics are so dissimilar from one another. By normalizing the values of all the data points, normalization ensures that all the characteristics are treated with the same value.

Min-Max normalization is one of the most common ways to normalize data. For every feature, the minimum value of that feature gets transformed into a 0, the maximum value gets transformed into a 1, and every other value gets transformed into a decimal between 0 and 1. For example, if the minimum value of a feature was 20, and the maximum value was 40, then 30 would be transformed to about 0.5 since it is halfway between 20 and 40. However, Min-Max normalization has one fairly significant downside. It does not handle outliers very well. For example, if you have 99 values between 0 and 40, and one value is 100, then the 99 values will all be transformed to a value between 0 and 0.4.

On the other hand, Z-score normalization is a strategy of normalizing data that avoids the outlier issue of Min-Max normalization. This method is working along with mean and standard deviation of the data instead of minimum and maximum in Min-Max normalization. Z-score normalization works in the way that if a value is exactly equal to the mean of all the values of the feature, it will be normalized to 0. If it is below the mean, it will be a negative number, and if it is above the mean, it will be a positive number. The size of those negative and positive numbers is determined by the standard deviation of the original feature. If the unnormalized data had a large standard deviation, the normalized values will be closer to 0.

In this instance, the Z-score normalization technique was applied to all of the features in the joint feature vector. We may use either the "zscore" function or the "normalize" function to produce a normalized joint feature vector in MATLAB. Both of these functions normalize the data with a center of 0 and a standard deviation of 1 in a reasonable manner.

Overall, normalizing the data is a key element of machine learning. If we neglect to normalize, one of our features may entirely overpower the rest, even if there is a superb dataset with a multitude of valuable characteristics. It is like tossing away practically all of the information and normalizing addresses this problem.

4.3 Feature Vector Concatenation

After all of the features have been obtained for each face image using both LBP and HOG approaches, the normalized feature vectors need to be concatenated in order to get a single normalized feature vector for each image.

The function "cat" had been used in MATLAB-based technique for this purpose and it allows to mix the LBP and HOG feature vectors. After that, the remaining operations will utilize the combined single feature vector that was created.

4.4 Matching Stage of the Proposed Method

In the matching phase, the extracted features are employed to build a system model that classifies test images into the correct category based on the learnt model that predicts whether or not the test images will be classified as correct recognitions or incorrect ones.

Given the ease of usage, Manhattan Distance is frequently employed in place of a completely hand-crafted classifier. Since it is written as a dynamic program, this classifier may use the same set of parameters regardless of the specifics of the data being analyzed. There is a comparison between the model that has been trained and the test image to determine how similar they are. With the facial features retrieved, the class of the test image is determined by the classifier's distance output. At the end, a test image is assigned to one of the correct recognition classes or one of the incorrect ones.

Chapter 5

EXPERIMENTAL WORKS AND RESULTS

Experiments utilizing different feature extractors are run on publicly accessible face recognition datasets to determine their overall recognition performance. In the sections that follow, details on these face recognition datasets are given. After that, many experimental designs are outlined. Discussion of experimental findings are presented in the last subsection.

5.1 Description of Datasets

In the experiments, three benchmark datasets are employed, namely CALFW, MLFW and FERET. A brief explanation of each dataset is provided in the subsequent sections of this chapter.

5.1.1 Cross-Age Labeled Faces in the Wild (CALFW)

Cross-Age Labeled Faces in the Wild (CALFW) [19] dataset is commonly used to evaluate the performance of face recognition algorithms. Cross-age face is a key difficulty in face identification that LFW does not pay much focus on, along with variable postures, illuminations, occlusions and moods. They create a Cross-Age LFW (CALFW) by locating and selecting 3,000 positive face pairings with age differences, thereby accounting for the intra-class variation produced by the ageing process. They also choose negative pairs that have the same racial and gender composition as the positive pairs in order to reduce the effect of attribute disparities across positive/negative combinations.

The dataset comprised of a set of images of faces that have been classified according to their age, gender, or other identifying characteristics. The collection includes images of people gathered from many places, including the Internet, and they represent a wide range of poses, expressions and lighting conditions.

The CALFW dataset was created to solve the issue of recognizing faces across age ranges. This is a challenging problem because faces change significantly with age, and the changes are not just limited to wrinkles, but also with the facial features shape and size.

The dataset includes more than 12 thousand images of according to 557 individuals, with age ranging from 16 to 62. This database has 6,000 unique facial combinations (50% positive and 50% negative pairings). The dataset includes both annotated and unannotated versions, where the annotated version includes age labels, while the unannotated version does not. These annotations are intended to be used for assessing how well various facial recognition algorithms can guess a person's age, while the unannotated version can be utilized for general face identification testing.

The CALFW dataset is, in general, a real blessing for scholars working on the challenge of cross-age face recognition, as it has a big variety of images of faces of varying ages. Some face image samples can be seen in Figure 9.

Figure 9: CALFW Dataset Samples [19]

5.1.2 Mask Labeled Faces in the Wild (MLFW)

Because of the ongoing COVID-19 epidemic, an increasing number of people are choosing to cover their faces with masks. This may cause existing facial recognition systems to see a significant drop in performance when trying to identify masked faces.

With the Cross-Age LFW (CALFW) database as a starting point, they developed a straightforward technique for establishing a new database, Masked LFW, to hold all of the masked faces generated automatically from unmasked faces (MLFW), to investigate the impact of masks on facial recognition models.

Images of human faces, annotated with labels for common facial features and masks, make up the Masked Labeled Faces in the Wild (MLFW) [20] dataset. The images in the dataset include individuals with face masks on, and the dataset is designed to be useful for facial analysis tasks, such as face alignment, facial expression analysis and facial feature extraction. This dataset may be used to test how well facial recognition and analysis algorithms perform when faced with masked faces.

The dataset contains exact 12,000 images of faces from a diverse set of individuals, with age ranging from 1 to 70. The dataset includes images of faces in a wide range of poses, expressions, lighting conditions and also includes a variety of facial landmarks and masks. Some samples with age variance can be seen in Figure 10.

It is worth noting that datasets related to face recognition with masks can be highly important in various applications such as surveillance, security and access control in times of pandemics or other situations that make wearing masks mandatory.

Figure 10: MLFW Dataset Samples [20]

5.1.3 The Facial Recognition Technology (FERET)

The FERET (Facial Recognition Technology) [21] dataset consists of face images that was developed by the United States government in the 1990s to test how well face recognition programs work. The dataset was created by the National Institute of Standards and Technology (NIST) and is among the most widely used datasets for evaluating facial recognition algorithms.

The FERET dataset contains over 14,000 images of faces from more than 1,000 individuals. Under controlled environments, the images in the dataset were captured, with the individuals posing in a neutral expression, facing the camera directly and with consistent lighting. The collection contains images of faces in a variety of positions, facial expressions and illumination situations and also includes a variety of facial landmarks.

Specifically, there are two distinct subsets of the dataset: the training set and the testing set. A total of 856 people is represented in the training set, whereas 115 are seen in the test set. The images in the dataset have been labeled with information such as identity, age and gender.

For this study, a total of 1000 face images from 250 different individuals were selected from the FERET dataset. Prior to the evaluation phase, all the images were preprocessed by cropping, resizing and other standard preprocessing techniques to ensure that the images are ready for evaluation. Some sample images are shown in Figure 11.

Figure 11: FERET Dataset Samples [21]

5.2 Experimental Methodology

The experiments were implemented using MATLAB 2022a using a computer with Intel i7 processor (10^{th} gen.) and 16 GB of RAM. The experiments are typically organized into four sections to further illustrate the performance. The first section involves training and testing with masked images of faces. The second section involves training with masked face images and testing with unmasked ones. The third experiment used unmasked and masked face images for training and evaluation respectively. The fourth section involves training and evaluating the approaches on unmasked facial images. The fifth section involves training and testing of all feature extraction approaches on a different unmasked face image datast.

All approaches are evaluated under identical conditions, using a similar amount of test and training images (50%-50%). In the following sections, we demonstrate the best findings after using four different feature extraction approaches.

5.3 Experiments using Different Datasets

The outcomes of the tests, which are carried out on several datasets, are as follows. Every feature extractor is applied on each database in five different experimental setups. Features from each and every train set are retrieved, and a handcrafted classifier is used to match those features. Information about datasets used in this thesis for five experimental setups are abridged in Table 1.

Table 1: Information of Datasets Used in Experimental Setups

Exp. Setup	Train Set	Train Image per Person	Total number of Train Images	Test Set	Test Image per Person	Total number of Test Images	Total number of Entities	Total number of Images
I	MLFW	1	500	MLFW	1	500	500	1000
II	MLFW	1	500	CALFW	1	500	500	1000
III	CALFW	1	500	MLFW	1	500	500	1000
IV	CALFW	1	500	CALFW	1	500	500	1000
V	FERET	2	500	FERET	2	500	250	1000

5.3.1 Experimental Setup I

In the first experimental setup, all feature extraction methods are employed for masked face recognition after being trained on images of masked faces. Moreover, images for this experimental setup had been chosen from the MLFW dataset.

Furthermore, the objects were split down the middle, with half used for training and the other half used for testing. Two masked face images are chosen for each of 500 different people, for a total of 1000 face images. Also, the final results had been measured in both cases so that we can train the program with the first image and test with the second one, or vice versa. Table 2 illustrates the observations.

Table 2: Recognition Rates Using the Experimental Setup I on MLFW Dataset

	Feature Extractor	Train Image (MLFW)	Test Image (MLFW)	Accuracy
1	PCA	1^{st}	2^{nd}	70.2 %
2	LBP	1^{st}	2^{nd}	93.2 %
3	HOG	1^{st}	2^{nd}	93.2 %
4	Proposed Method	1^{st}	2^{nd}	94.2 %
5	PCA	2^{nd}	1^{st}	68.8 %
6	LBP	2^{nd}	1^{st}	93.6 %
7	HOG	2^{nd}	1^{st}	92.8 %
8	Proposed Method	2^{nd}	1^{st}	94.8 %

It is apparent that in this study, the results from the Proposed, LBP, and HOG methods are all above 90% and very similar to each other, however, the PCA method does not perform as well as these three methods. The proposed method has the highest recognition rate, with a 1% advantage over the other methods. In this experimental scenario, HOG approach requires the least time with around 5 seconds, while PCA method requires the maximum time with approximately 380 seconds. In addition, proposed method requires close to 22 seconds to be trained and evaluate the entire test images.

5.3.2 Experimental Setup II

In the second part of the experiment, all feature extraction methods are used to recognize unmasked faces after being trained on images of faces with masks. Masked face images are chosen from the MLFW dataset in this case scenario, while unmasked face images are chosen from the CALFW dataset.

Also, images are split evenly so that half are utilized for learning and the other half for testing. One masked face image is chosen for each of the 500 different people to train the program, and one unmasked face image is used for testing, for a total of 1000 face images. Table 3 displays the findings.

Table 3: Recognition Rates Using the Experimental Setup II on MLFW and CALFW Datasets

	Feature Extractor	Train Image (MLFW)	Test Image (CALFW)	Accuracy
1	PCA	Masked	Unmasked	45.6 %
2	LBP	Masked	Unmasked	57.6 %
3	HOG	Masked	Unmasked	57.6 %
4	Proposed Method	Masked	Unmasked	64.8 %

It is clear that in this study, the proposed approach outperforms the others by roughly 7% in terms of recognition rate. Following the proposed method, LBP and HOG methods have the same recognition rate, and finally, PCA method is at the bottom with a recognition rate under 50%. In this experimental scenario, HOG technique takes the shortest amount of time, roughly 6 seconds, while PCA method requires the most amount of time, approximately 380 seconds. In addition, it takes around 37 seconds for the proposed method to be trained and analyze all test images.

5.3.3 Experimental Setup III

In the third phase of the experiment, all feature extraction algorithms were utilized to distinguish masked faces following training on images of unmasked faces. In this situation, face images with masks are selected from the MLFW dataset, whereas face images without masks are selected from the CALFW dataset.

In addition, 50% of the images are used for training and 50% for evaluation. One unmasked face image for each of the 500 unique individuals is used to train the algorithm, and, one masked face image for testing for a total of 1000 face images. Table 4 provides a summary of the results.

Table 4: Recognition Rates Using the Experimental Setup III on CALFW and MLFW Datasets

	Feature Extractor	Train Image (CALFW)	Test Image (MLFW)	Accuracy
1	PCA	Unmasked	Masked	30.4 %
2	LBP	Unmasked	Masked	58.0 %
3	HOG	Unmasked	Masked	58.0 %
4	Proposed Method	Unmasked	Masked	58.6 %

In the third experimental setup, when algorithms initially trained by unmasked imagery and then evaluated on masked ones, the proposed method, along with LBP and HOG methods, produce the best results with a recognition rate of near 60%. PCA method, in this case, does not perform well. In this experimental setup, HOG technique takes the least amount of time, roughly 9 seconds, while the PCA method requires the most, approximately 170 seconds. In addition, it takes around 34 seconds for the proposed method to be trained and analyze all test images.

5.3.4 Experimental Setup IV

Following training on images of unmasked faces, all feature extraction methods are used to distinguish unmasked faces in the fourth phase of the experiments. Face images from the CALFW dataset are used in this case.

Furthermore, fifty percent of the images are used as train set and the other fifty percent for assessment. For each of the 500 unique individuals used to train and test the algorithm, two unmasked face images are chosen for a total of 1000 face images. Table 5 summarizes the findings.

Table 5: Recognition Rates Using the Experimental Setup IV on CALFW Dataset

	Feature Extractor	Train Image (CALFW)	Test Image (CALFW)	Accuracy
1	PCA	1st	2nd	33.6 %
2	LBP	1st	2nd	47.0 %
3	HOG	1st	2nd	46.8 %
4	Proposed Method	1st	2nd	52.2 %
5	PCA	2nd	1st	36.6 %
6	LBP	2nd	1st	49.2 %
7	HOG	2nd	1st	45.6 %
8	Proposed Method	2nd	1st	52.8 %

Unfortunately, due to the lack of high-quality, unprocessed, not aligned and also age variant images in the CALFW dataset, in the case of regular facial recognition, overall performance is poor in the beginning of this experimental setup. This experimental setup is improved following extensive pre-processing on 1000 images selected from the CALFW dataset, including aligning, cropping the face region and histogram equalization. In this experimental setup, it is obvious that the proposed method outperforms the competition. Additionally, while PCA approach takes around 225 seconds, HOG methodology takes just about 30 seconds in the testing configuration. Additionally, training and analyzing all test images using the proposed method takes only 119 seconds.

5.3.5 Experimental Setup V

After being trained on images of faces without masks, all the feature extraction techniques are applied to recognize unmasked faces but with a different set of data in the fifth part of this experiment. The images utilized in this instance are from the FERET dataset. In this experimental setup, the testing is conducted on a total number of 1000 unmasked face images that contain 4 face images per person. At first, the first two images are used for training while the other two of them have been tested through all feature extraction methods after that the training set and test set has been changed with each other to assess the performance of all feature extraction methods in different case scenarios.

Additionally, the image samples are divided into two comparable halves, one for training while the other is for evaluation. For each 250 unique individuals chosen for training and testing the algorithms, all the images are pre-processed before conducting the testing. Table 6 summarizes the outcomes of this study.

Table 6: Recognition Rates Using the Experimental Setup V on FERET Dataset

	Feature Extractor	Train Image (FERET)	Test Image (FERET)	Accuracy
1	PCA	1^{st} - 2^{nd}	3^{rd} – 4^{th}	92.6 %
2	LBP	1^{st} - 2^{nd}	3^{rd} – 4^{th}	98.4 %
3	HOG	1^{st} - 2^{nd}	3^{rd} – 4^{th}	94.8 %
4	Proposed Method	1^{st} - 2^{nd}	3^{rd} – 4^{th}	98.4 %
5	PCA	3^{rd} – 4^{th}	1^{st} - 2^{nd}	89.2 %
6	LBP	3^{rd} – 4^{th}	1^{st} - 2^{nd}	97.6 %
7	HOG	3^{rd} – 4^{th}	1^{st} - 2^{nd}	96.6 %
8	Proposed Method	3^{rd} – 4^{th}	1^{st} - 2^{nd}	98.0 %

In the fifth experimental setup, the real performance of the techniques that we have used can be evaluated. The approaches are applied on the FERET dataset for normal face recognition without face masks. Additionally, the image sets used for training and evaluation are altered for observing the performance under different conditions. It is clear that all the methods have acceptable recognition percentages of above 90%, however, the performance of LBP and proposed methods in some cases are nearly 100%. Comparatively, in this experimental setup, PCA technique takes roughly 140 seconds whereas HOG method takes just 10 seconds. In addition, the proposed method only requires 11 seconds to train and analyze all test images.

5.4 Comparison with State-of-the-Art Models

At the conclusion of these experimental activities, the best results relevant to this thesis are compared with various state-of-the-art models. These open sourced state-of-the-art deep face recognition methods are as follows: (1) ResNet50 model trained on a private Asia face dataset [44] with ArcFace [45], (2) ResNet50 model trained on CASIAWebFace database [46] with ArcFace [45], (3) ResNet50 model trained on VGGFace2 database [47] with ArcFace [45], (4) ResNet100 model trained on MS1MV2 database [48] refined by insightface with ArcFace [45], (5) ResNet100 model trained on MS1MV2 database [48] with CurricularFace [49], (6) ResNet100 model trained on MS1MV2 database [48] with SFace [50].

All of these state-of-the-art methods are compared to the proposed method in this thesis in terms of accuracy using Cross-age LFW (CALFW) and Masked LFW (MLFW) datasets. It is important to note that, for the proposed method, CALFW and MLFW datasets are used in different cases in order to train the model, while for those deep learning methods, different benchmarks using various loss functions were used for this purpose. The comparison results are shown in Table 7.

Table 7: Comparison of the Proposed Method with the State-of-the-Art

Reference	Publication Year	Method	Train Set	Accuracy on CALFW	Accuracy on MLFW
Wang et al. [44]	2021	ResNet50 (ArcFace)	Private-Asia	91.12 %	74.85 %
Yi et al. [46]	2014	ResNet50 (CosFace)	Casia-WebFace	92.43 %	82.87 %
Cao et al. [47]	2018	ResNet50 (ArcFace)	VGGFace2	93.72 %	85.02 %
Guo et al. [48]	2016	ResNet100 (ArcFace)	MS1MV2	95.83 %	90.13 %
Guo et al. [48]	2016	ResNet100 (Curricularface)	MS1MV2	95.97 %	90.60 %
Guo et al. [48]	2016	ResNet100 (SFace)	MS1MV2	95.83 %	90.63 %
Proposed Method	2023	Feature-level Fusion of HOG and LBP	MLFW	64.80 %	94.80 %
			CALFW	52.80 %	58.60 %

In the masked face recognition task, the proposed method in this thesis has better performance compared to the top 6 deep learning methods, while in the case of unmasked face recognition on the CALFW dataset using masked face images on the MLFW dataset for training, the performance is lower than the other deep learning methods. It is critical to note that the MLFW dataset was used as a training set in this thesis, as opposed to other high computational cost and time training sets that have been used for deep learning models. Otherwise, using the CALFW dataset as the training set, the performance of the proposed method for both face recognition tested on the CALFW dataset and masked face recognition tested on the MLFW dataset is over 50%.

To wrap up, it is important to note that in this thesis, two datasets are used for training and testing in different case scenarios instead of those used with deep learning models using different loss functions to decrease the computation time and perform the task of face recognition with the least computation cost.

Chapter 6

CONCLUSIONS

6.1 Conclusions

In this thesis, an in-depth examination of various analyses of facial recognition algorithms are undertaken by measuring the efficiency of four methods for extracting features, namely Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), Principal Component Analysis (PCA) and a custom-designed proposed method using the feature-level fusion of LBP and HOG methods. Furthermore, a hand-crafted classifier is applied to assess the precision of the recognition procedure. The experiments are conducted using three benchmark datasets namely, the Cross-Age LFW (CALFW), Masked Labeled in the Wild (MLFW) and the Facial Recognition Technology (FERET) datasets. These datasets are selected because they are widely employed in the area of facial recognition and offer a trustworthy assessment of the performance of the systems under investigation. The findings are analyzed and compared to determine the effectiveness of each method. The study also aims to identify any potential limitations and areas for further improvement in the field of face recognition.

This study indicates that when it comes to recognizing masked faces using a training set of masked face images, the proposed method outperformed the other feature extraction methods, while Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP) methods have nearly similar performance to the proposed method. In the case of recognizing masked faces using a training set of unmasked face images, the proposed method has the highest performance, followed by Local Binary Patterns and Histogram of Oriented Gradients methods, which have comparable accuracy levels. When it comes to recognizing unmasked faces using a training set of masked faces, the proposed method again is the most accurate one, and the performance of the Local Binary Patterns and Histogram of Oriented Gradients methods are equivalent. After these tests, for latter two studies, the test is conducted on unmasked face image datasets to evaluate the accuracy of these feature extractors in normal face recognition scenarios. It is observed that in the instance of regular facial recognition using the Cross-Age Labeled in the Wild (CALFW) dataset, overall performance in the beginning is poor because the Cross-Age Labeled in the Wild (CALFW) dataset lacked high-quality and pre-processed images. After some improvement following extensive pre-processing on 1000 images selected from the CALFW dataset, including aligning, cropping the face region, and histogram equalization, it can be seen that the proposed method outperforms the other feature extraction methods. Finally, in the last experiment to assess the actual performance of these feature extractors in normal face recognition, The Facial Recognition Technology (FERET) dataset is used. Here, though, the proposed and Local Binary Patterns methods are found to be the best, and all the methods have an accuracy of 90% or higher, except for the PCA method.

6.2 Future Works

Machine learning and deep learning methodologies are constantly evolving. Regarding future work, it would be beneficial to use more benchmark unmasked and masked face image datasets that include a variety of illumination angles and rotation variations, as well as incorporating other feature extraction methods based on deep learning techniques. Additionally, classifiers play a crucial role in face recognition applications, so incorporating a neural network-based classifier could help to push the boundaries even further.

REFERENCES

[1] F. Ghahramani, "Face Recognition: An Engineering Approach" by Farshad Ghahramani, Jan. 01, 2015. https://scholarworks.sjsu.edu/etd_theses/4635

[2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[3] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," Proceedings. International Conference on Image Processing, 2002, pp. I-I, doi: 10.1109/ICIP.2002.1038171.

[4] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711-720, July 1997, doi: 10.1109/34.598228.

[5] C. -H. Lin and B. -F. Wu, "Domain Adapting Ability of Self-Supervised Learning for Face Recognition," 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 2021, pp. 479-483, doi: 10.1109/ICIP42928.2021.9506677.

[6] A. Mian, "Unsupervised learning from local features for video-based face recognition," 2008 8th IEEE International Conference on Automatic Face & Gesture Recognition, Amsterdam, Netherlands, 2008, pp. 1-6, doi: 10.1109/AFGR.2008.4813310.

[7] G. F. Ling, P. Y. Han, K. E. Yee and O. S. Yin, "Face recognition via semi-supervised discriminant local analysis," 2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala Lumpur, Malaysia, 2015, pp. 292-297, doi: 10.1109/ICSIPA.2015.7412207.

[8] L. Yadav, R. K. Yadav and V. Kumar, "An Efficient Approach towards Face Recognition using Deep Reinforcement Learning, Viola Jones and K-nearest neighbor," 2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems (ACCESS), Ernakulam, India, 2021, pp. 112-117, doi: 10.1109/ACCESS51619.2021.9563326.

[9] F. Liu, Y. Ding, S. Yang and F. Xu, "Patch Based Semi-Supervised Linear Regression for Single Sample Face Recognition," 2017 IEEE Third International Conference on Multimedia Big Data (BigMM), Laguna Hills, CA, USA, 2017, pp. 62-65, doi: 10.1109/BigMM.2017.41.

[10] S. Ongkittikul, J. Suwatcharakulthorn, K. Chutisowan and K. Ratanarangsank, "Convolutional Multinomial Logistic Regression for Face Recognition," 2020 8th International Electrical Engineering Congress (iEECON), Chiang Mai, Thailand, 2020, pp. 1-4, doi: 10.1109/iEECON48109.2020.246699.

[11] W. -S. Yang, C. -W. Tsai, K. -M. Cho, C. -S. Yang, S. -J. Lin and M. -C. Chiang, "A semi-supervised support vector machine-based algorithm for face recognition," 2009 IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, 2009, pp. 1609-1614, doi: 10.1109/ICSMC.2009.5346743.

[12] R. Cohn and E. Holm, "Unsupervised machine learning via transfer learning and k-means clustering to classify materials image data, " arXiv.org, Jul. 16, 2020. https://arxiv.org/abs/2007.08361v1

[13] A. Ruiz-Garcia, V. Palade, I. Almakky and M. Elshaw, "Deep Q-Learning for Illumination and Rotation Invariant Face Detection," 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 2019, pp. 1-8, doi: 10.1109/IJCNN.2019.8851861.

[14] S. R. Kundeti, D. Parmar, A. S. MV and D. Gautam, "Landmark Detection in 3D Medical Images Using Reinforcement Learning," 2020 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), Bengaluru, India, 2020, pp. 42-46, doi: 10.1109/CCEM50674.2020.00019.

[15] T. F. Cootes, K. Walker and C. J. Taylor, "View based active appearance models, " IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble, France, pp. 227-23, Mar 2000.

[16] G. E. Rani, S. M, M. P. Suresh, M. Abhiram, K. J. Surya and B. Y. A. N. Kumar, "Face Recognition Using Principal Component Analysis," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2022, pp. 932-936, doi: 10.1109/ICACITE53722.2022.9823434.

[17] O. S. Kulkarni, S. M. Deokar, A. K. Chaudhari, S. S. Patankar and J. V. Kulkarni, "Real Time Face Recognition Using LBP Features," 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA), 2017, pp. 1-5, doi: 10.1109/ICCUBEA.2017.8463886.

[18] H. Ahamed, I. Alam and M. M. Islam, "HOG-CNN Based Real Time Face Recognition," 2018 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE), 2018, pp. 1-4, doi: 10.1109/ICAEEE.2018.8642989.

[19] T. Zheng, W. Deng, and J. Hu, "Cross-age LFW: A database for studying cross-age face recognition in unconstrained environments," CoRR, vol. abs/1708.08197, 2017.

[20] C. Wang, H. Fang, Y. Zhong, and W. Deng, "MLFW: A Database for Face Recognition on Masked Faces, " In: et al. Biometric Recognition. CCBR 2022. Lecture Notes in Computer Science, vol 13628. Springer, Cham. 2022. doi: 10.1007/978-3-031-20233-9_18.

[21] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," NISTIR 6264 and IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, No 10, Oct. 2000. doi: doi.org/10.1016/S0262-8856(97)00070-X

[22] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1991, pp. 586-591, doi: 10.1109/CVPR.1991.139758.

[23] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711-720, July 1997, doi: 10.1109/34.598228.

[24] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, July 2002, doi: 10.1109/TPAMI.2002.1017623.

[25] T. Cover and P. Hart, "Nearest neighbor pattern classification," in IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21-27, January 1967, doi: 10.1109/TIT.1967.1053964.

[26] C. Shen, J. Kim and L. Wang, "Scalable Large-Margin Mahalanobis Distance Metric Learning," in IEEE Transactions on Neural Networks, vol. 21, no. 9, pp. 1524-1530, Sept. 2010, doi: 10.1109/TNN.2010.2052630.

[27] T. -J. Chin and D. Suter, "Incremental Kernel Principal Component Analysis," in IEEE Transactions on Image Processing, vol. 16, no. 6, pp. 1662-1674, June 2007, doi: 10.1109/TIP.2007.896668.

[28] H. Roy, D. Bhattacharjee and O. Krejcar, "Interpretable Local Frequency Binary Pattern (LFrBP) Based Joint Continual Learning Network for Heterogeneous Face Recognition," in IEEE Transactions on Information Forensics and Security, vol. 17, pp. 2125-2136, 2022, doi: 10.1109/TIFS.2022.3179951.

[29] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[30] A. Howard et al., "Searching for MobileNetV3," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 1314-1324, doi: 10.1109/ICCV.2019.00140.

[31] G. B. Huang, M. Mattar, T. Berg and E. Learned-Miller., "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments," Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[32] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller and E. Brossard, "The MegaFace Benchmark: 1 million Faces for Recognition at Scale," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 4873-4882, doi: 10.1109/CVPR.2016.527.

[33] Z. Wang, B. Huang, G. Wang, P. Yi and K. Jiang, "Masked Face Recognition Dataset and Application," in IEEE Transactions on Biometrics, Behavior, and Identity Science, doi: 10.1109/TBIOM.2023.3242085.

[34] O. Due Trier, A. Jain and T. Taxt, "Feature extraction methods for character recognition - A survey", Pattern Recognition, vol. 29, no. 4, pp. 641-662, 1996.

[35] D. A. Lisin, M. A. Mattar, M. B. Blaschko, E. G. Learned-Miller and M. C. Benfield, "Combining Local and Global Image Features for Object Class Recognition," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops, San Diego, CA, USA, 2005, pp. 47-47, doi: 10.1109/CVPR.2005.433.

[36] M. N, A. Nayak, P. N R and V. A, "Performance analysis of various distance measures for PCA based Face Recognition," 2015 National Conference on Recent Advances in Electronics & Computer Engineering (RAECE), Roorkee, India, 2015, pp. 130-133, doi: 10.1109/RAECE.2015.7510240.

[37] L. Greche, M. Jazouli, N. Es-Sbai, A. Majda and A. Zarghili, "Comparison between Euclidean and Manhattan distance measure for facial expressions classification," 2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), Fez, Morocco, 2017, pp. 1-4, doi: 10.1109/WITS.2017.7934618.

[38] H.V. Nguyen and L. Bai, "Cosine Similarity Metric Learning for Face Verification. " In: Kimmel, R., Klette, R., Sugimoto, A. (eds) Computer Vision – ACCV 2010. ACCV 2010. Lecture Notes in Computer Science, vol 6493. Springer, Berlin, Heidelberg. doi: 10.1007/978-3-642-19309-5_55

[39] A. M. Fraser, N. W. Hengartner, K. R. Vixie and B. E. Wohlberg, "Incorporating invariants in Mahalanobis distance-based classifiers: application to face recognition," Proceedings of the International Joint Conference on Neural Networks, 2003., Portland, OR, USA, 2003, pp. 3118-3123 vol.4, doi: 10.1109/IJCNN.2003.1224070.

[40] L. Li, P. Fieguth, and G. Kuang, "Generalized Local Binary Patterns for Texture Classification" In Jesse Hoey, Stephen McKenna and Emanuele Trucco, Proceedings of the British Machine Vision Conference, pages 123.1-123.11. BMVA Press, 2011. doi: 10.5244/C.25.123

[41] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893 vol. 1, doi: 10.1109/CVPR.2005.177.

[42] J. Liu, "Image Retrieval based on Bag-of-Words model, " arXiv.org, Apr. 18, 2013. https://arxiv.org/abs/1304.5168v1 (accessed Jan. 11, 2023).

[43] J. Müller, A. Fregin and K. Dietmayer, "Disparity Sliding Window: Object Proposals from Disparity Images," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 2018, pp. 5777-5784, doi: 10.1109/IROS.2018.8593390.

[44] Q. Wang, P. Zhang, H. Xiong, and J. Zhao, "Face.evoLVe: A High-Performance Face Recognition Library," arXiv.org, Jul. 19, 2021. https://arxiv.org/abs/2107.08621v4

[45] J. Deng, J. Guo, N. Xue and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 4685-4694, doi: 10.1109/CVPR.2019.00482.

[46] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning Face Representation from Scratch, " arXiv.org, Nov. 28, 2014. https://arxiv.org/abs/1411.7923v1

[47] Q. Cao, L. Shen, W. Xie, O. M. Parkhi and A. Zisserman, "VGGFace2: A Dataset for Recognising Faces across Pose and Age," 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xi'an, China, 2018, pp. 67-74, doi: 10.1109/FG.2018.00020.

[48] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-Celeb-1M: A Dataset and Benchmark for Large-Scale Face Recognition, " Sep. 17, 2016. https://link.springer.com/chapter/10.1007/978-3-319-46487-9_6

[49] Y. Huang et al., "CurricularFace: Adaptive Curriculum Learning Loss for Deep Face Recognition," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 5900-5909, doi: 10.1109/CVPR42600.2020.00594.

[50] Y. Zhong, W. Deng, J. Hu, D. Zhao, X. Li and D. Wen, "SFace: Sigmoid-Constrained Hypersphere Loss for Robust Face Recognition," in IEEE Transactions on Image Processing, vol. 30, pp. 2587-2598, 2021, doi: 10.1109/TIP.2020.3048632.